

# Aula 7: Laços DO-WHILE e FOR

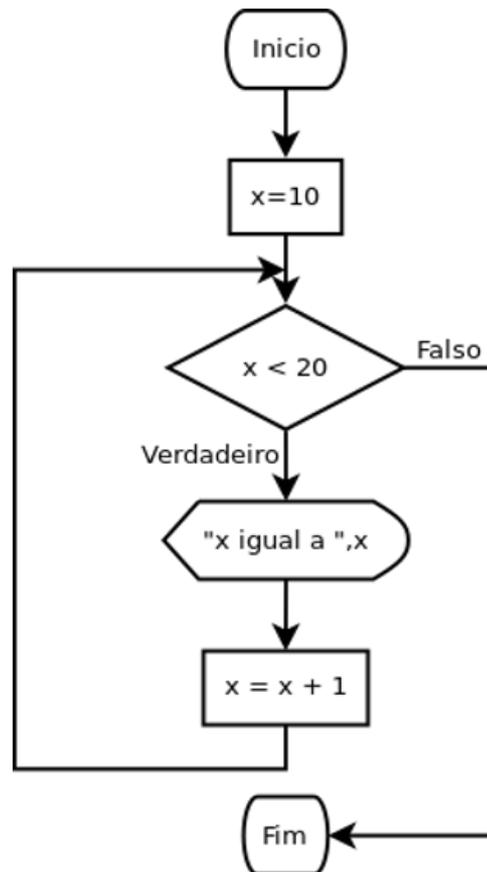
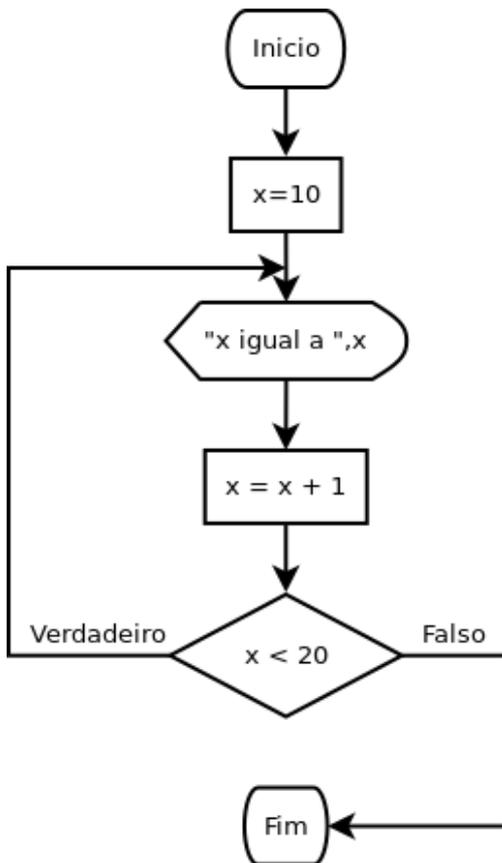
Prof. Sérgio Montazzoli Silva  
smsilva@uel.br

# Sumário

- Nesta aula:
  - Laço DO-WHILE
  - Laço FOR

# While e Do-While (Fluxograma)

- Qual a diferença entre estes fluxogramas?



O que acontece se alterarmos a condição  $x < 20$  para  $x < 10$  ?

# Laço DO-WHILE

- Laço DO-WHILE significa FAÇA-ENQUANTO
- É semelhante ao laço WHILE, porém a avaliação é feita após a execução do laço, e não antes
- Útil quando a primeira execução obrigatoriamente deve ser feita.
- Sintaxe em C:  

```
do {  
    ... seu código vai aqui ...  
} while (expressão);
```

# Exemplo

```
#include <stdlib.h>
#include <stdio.h>

int main() {

    char entrada;

    do {
        scanf(" %c",&entrada);
        printf("Tecla digitada: %c\n",entrada);
    } while (entrada == 'q');

}
```

- O que este programa faz?

# Comparação

- Identifique o erro na versão com WHILE

```
#include <stdlib.h>
#include <stdio.h>

int main() {
    char entrada;

    do {
        scanf(" %c",&entrada);
        printf("Tecla digitada: %c\n",entrada);
    } while (entrada == 'q');
}
```

```
#include <stdlib.h>
#include <stdio.h>

int main() {
    char entrada;

    while (entrada == 'q') {
        scanf(" %c",&entrada);
        printf("Tecla digitada: %c\n",entrada);
    }
}
```

- Valor da variável "entrada" não foi especificado antes da comparação
- Por que no laço DO-WHILE não é preciso especificar?

# Exercício DO-WHILE

- Crie um programa simples que repetidamente leia um número inteiro e termine apenas quando o número lido for maior do que 300.

# Laço FOR

- Utiliza 3 expressões para repetir algo por um número fixo de vezes
- Necessita de:
  - Uma expressão de **inicialização**
  - Uma expressão de **teste**
  - Uma expressão de **incremento**
- Em C:  
**for (inicialização; teste; incremento) {**  
    ... seu código aqui ...  
**}**

# Exemplo

- Contagem de 0 até 9

```
#include <stdlib.h>
#include <stdio.h>
```

```
int main() {

    int contador;

    for (contador = 0; contador < 10; contador++) {
        printf("%d\n", contador);
    }

}
```

# WHILE vs. FOR

- Um laço FOR pode facilmente ser convertido em um laço WHILE, basta recriar as expressões de inicialização, teste e incremento nos pontos certos do seu código.
- No programa abaixo, onde estão estas expressões?

```
int main() {  
    int contador;  
    contador = 0;  
    while (contador < 10) {  
        printf("%d\n", contador);  
        contador++;  
    }  
}
```

Diagram illustrating the components of a while loop in the provided code:

- inicialização** (Initialization): Points to the line `contador = 0;`
- teste** (Test): Points to the condition `(contador < 10)` in the while loop header.
- incremento** (Increment): Points to the line `contador++;` inside the while loop body.

# Exemplos de Expressões

- Supondo que já existam duas variáveis inteiras  $i$  e  $j$ :

	Inicialização	Teste	Incremento
(1)	$i=0$	$i < 10$	$i++$
(2)	$i=0$	$i < 10$	$i = i + 1$
(3)	$i=100$	$i \neq 130$	$i = i + 2$
(4)	$i=0, j=0$	$(i+j) < 10$	$i++, j++$

- Que valores  $i$  e  $j$  irão assumir para cada iteração de FOR com as expressões da tabela?

# Outro Exemplo

- Quais valores serão impressos pela saída do seguinte programa?

```
#include <stdlib.h>
#include <stdio.h>

int main() {

    int i,j;

    for (i=0,j=-10; i > j; i++, j=j+2) {
        printf("Valor de i: %d, valor de j: %d\n",i,j);
    }

}
```

# Exercícios FOR

- Usando um laço do tipo FOR, crie um contador que inicie em 25 termine em 50, mostrando todos os números inteiros neste intervalo.
- Converta o programa abaixo, que usa WHILE, para FOR.

```
int main() {  
    int contador;  
    contador = 20;  
    while (contador > 0) {  
        printf("contador: %d\n", contador);  
        contador--;  
    }  
}
```

# Exercícios FOR

- Dada a seguinte equação:

$$s = \sum_{i=0}^{12} i \cdot (i + 2)$$

crie um programa imprima o resultado  $s$  da equação usando FOR.